UNIVERSITY OF THE WITWATERSRAND, JOHANNESBURG

University of the Witwatersrand, Johannesburg
School of Computer Science and Applied Mathematics

Course Outline*

Course code: COMS3008A
Course description: Parallel Computing. From smartphones, multi-core CPUs, many-core GPUs, to the world's largest supercomputers, parallel processing is a key player in the modern computing. In a simple way, parallel computing is about simultaneously using multiple compute resources to solve a computational problem. Parallel computing entails both hardware and software approaches. It is a result of the evolution and interaction of the developments of computer architecture, technology, and applications. This course involves introducing students to the fundamentals of parallel computing. This includes the basics of parallel computer architecture and parallel programming techniques necessary to effectively utilize a variety of parallel computers. While the course focuses on programming shared memory parallel computers, it also gives a light introduction to programming distributed memory systems. This course has a hands-on emphasis to understand the basics and myths of parallel computing.
Course prerequisite/prior knowledge

- Familiarity with programming in C/C++.
- Familiarity with the basics of data structures, algorithms, analysis of algorithms, and computer architectures.

Offered: Semester two
Course coordinator: Hairong Wang (email: `hairong.bau@wits.ac.za`)
Lecturer: Hairong Wang
Tutors: Kimessha Paupamah (head tutor, email: 1038238@students.wits.ac.za), Londani Zuma, David Wright, Jonathan Sibindi, Daniel Yazbek.
Course contents: The course covers the following topics:

- Part I: Foundations
  - Overview of parallel systems
  - Principles of parallel algorithm design
- Part II: Parallel programming
  - Programming shared memory multi-processors
  - Programming distributed memory system
- Selected parallel algorithms and their applications

Course aims: The aim of this course is to enable students to design efficient parallel programs for solving computational problems on a variety of parallel computers including shared memory and distributed memory systems.

*This course outline is subject to change. Changes, if any, will be announced in class.

Course objectives: The objectives of this course are for students to learn

- the fundamentals of parallel computing as they relate to solving computational problems using efficient parallel processing;
- parallel programming techniques for shared memory and distributed memory systems, as well as for graphics processing units;
- solving compute- or data-intensive computational problems using parallel processing;
- analyzing the performance of parallel algorithms with emphasis on performance gains.

Outcome: On completion of this course, students should have a good understanding of the fundamentals of parallel computing, principles of parallel algorithm design, and the basic hardware and software aspects in parallel computing; students should also be able to develop efficient parallel programs for solving computational problems for various parallel computers, such as a basic multi-core processor, a heterogeneous system with graphics processing units, or a multi-node cluster.

Teaching methods: Lectures will be delivered in traditional classroom style plus electronic slides display. Optionally, lectures could also be delivered remotely, where lectures will be pre-recorded and made available to students. All the course notes and slides being used during the class, lab tutorials and other notices will be placed on Sakai course site.

Schedule: There will be one double lecture weekly, and one afternoon lab session bi-weekly. See the table below.

Lecture	Thursday 8:00 am - 9:45 am
Lab	Tuesday 14:15 pm - 17:00 pm (COMS3008A/COMS3011A)
Consultation	Thursday 9:00 am - 10:00 am (valid during remote teaching)

Course assessment: See the table below.

Assessment	Weight (%)
Quiz (4 to 5)	25
Test (1)	20
Project	15
Exam	40

Textbook/Main references: There is no required textbook for this course. However, we use the following textbooks as major references.

- Introduction to Parallel Computing: From Algorithms to Programming on State-of-the-Art Platforms, by Roman Trobec, Boštjan Slivnik, Patricio Bulić, Borut Robič, Springer, 2018,
- Programming Massively Parallel Processors: A Hands-on Approach, second edition, by David B. Kirk and Wen-mei W. Hwu, Morgan Kaufmann Publishers Inc., 2013.
- Introduction to Parallel Computing, second edition, by Ananth Grama, Anshul Gupta, George Karypis, Vipin Kumar. Addison Wesley Publisher, 2003. (`https://www-users.cs.umn.edu/~karypis/parbook/`)

Other resources:

- Introduction to Parallel Computing, Blaise Barney, Lawrence Livermore National Laboratory. (`https://computing.llnl.gov/tutorials/parallel_comp/`)

Policies:

- Academic honesty should be strictly practised by all students, especially for the programming tasks in an assignment or a project.
- All students are expected to adhere to a standard of academic conduct, which demonstrates respect for themselves, their fellow students, and the educational mission of the University.
- All students are deemed by the University to understand that if they are found responsible for academic misconduct, they will be subject to the academic misconduct procedures, as outlined in the Students Handbooks.